

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-55434

(43)公開日 平成10年(1998) 2月24日

(51)Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 T 1/20			G 0 6 F 15/66	K
G 0 6 F 13/00	3 5 7		13/00	3 5 7 Z

審査請求 未請求 請求項の数7 O L (全 18 頁)

(21)出願番号 特願平8-212801

(22)出願日 平成8年(1996) 8月12日

(71)出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂二丁目17番22号

(72)発明者 平田 和貴

神奈川県足柄上郡中井町境430 グリーン

テクなかい 富士ゼロックス株式会社内

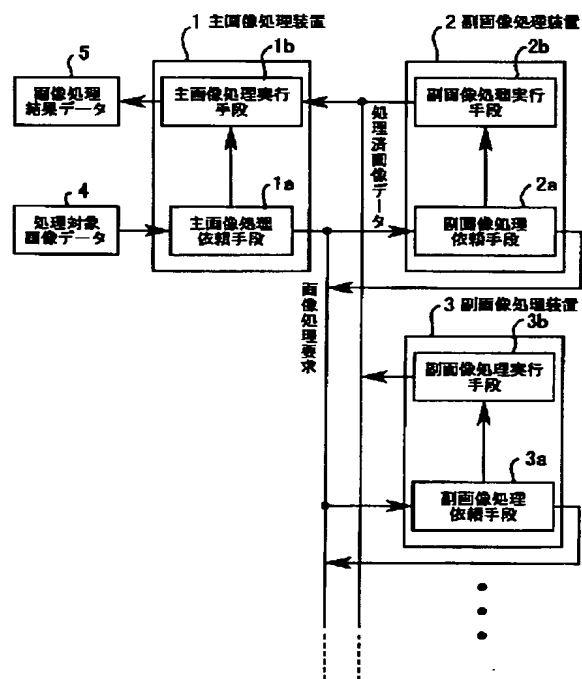
(74)代理人 弁理士 服部 毅藏

(54)【発明の名称】 画像処理システム

(57)【要約】

【課題】 スケジューリングのような複雑な処理を行わずに、画像データの分散処理を行う。

【解決手段】 ユーザが、処理対象画像データ4を主画像処理装置1に入力すると、主画像処理依頼手段1aにより、処理対象画像データの一部が副画像処理装置2、3に渡される。画像データを受けとった副画像処理装置内の副画像処理依頼手段は、さらに画像データの一部を複製し、それを他の副画像処理装置に渡す。そして、主画像処理実行手段1bと副画像処理実行手段2b、3bは、自己が受け取った画像データを処理する。処理済画像データは、副画像処理装置2、3から主画像処理装置1に渡される。主画像処理実行手段1bは、副画像処理装置2、3から受け取った処理済画像データと、自己が処理した画像データとを合成し、画像処理結果データ5として出力する。



【特許請求の範囲】

【請求項1】 画像データを分散して処理する画像処理システムにおいて、

入力された処理対象画像データの一部を複製し、複製した画像データの画像処理要求を出力する主画像処理依頼手段と、前記処理対象画像データに所定の処理を行うとともに、処理済画像データを受け取ると、受け取った処理済画像データと自己の処理した画像データとを合成し、所定のタイミングで画像処理結果データとして出力する主画像処理実行手段と、を有する主画像処理装置と、

画像処理要求を受け取ると、処理が依頼された画像データの一部を複製し、複製した画像データの画像処理要求を出力する副画像処理依頼手段と、処理が依頼された画像データに対して所定の処理を行い、処理済画像データとして前記主画像処理実行手段に転送する副画像処理実行手段と、を有する複数の副画像処理装置と、を有することを特徴とする画像処理システム。

【請求項2】 前記主画像処理実行手段は、処理対象画像データが入力された時から所定の時間が経過した際には、その時点で生成されている処理済画像データを、画像処理結果データとして出力することを特徴とする請求項1記載の画像処理システム。

【請求項3】 前記副画像処理実行手段は、画像処理要求を受け取ってから所定の時間が経過した際には、その時点で処理されている画像データを、処理済画像データとして前記主画像処理実行手段に転送することを特徴とする請求項1記載の画像処理システム。

【請求項4】 前記主画像処理依頼手段は、処理対象画像データの前半部分の複製又は後半部分の複製を生成し、前半部分の画像データの画像処理要求又は後半部分の画像データの画像処理要求を出力し、前記副画像処理依頼手段は、処理が依頼された画像データの前半部分の複製又は後半部分の複製を生成し、前半部分の画像データの画像処理要求又は後半部分の画像データの画像処理要求を出力する、ことを特徴とする請求項1記載の画像処理システム。

【請求項5】 前記主画像処理実行手段は、前記処理対象画像データのうち、前記主画像処理依頼手段により複製されていない部分から画像処理を行い、前記副画像処理実行手段は、処理が依頼された画像データのうち、前記副画像処理依頼手段により複製されていない部分の画像処理を行う、ことを特徴とする請求項1記載の画像処理システム。

【請求項6】 前記副画像処理依頼手段は、所定の時間間隔で、処理が依頼された画像データのうちの未処理の部分の一部を複製し、複製した画像データの画像処理要求を繰り返し出力することを特徴とする請求項1記載の画像処理システム。

【請求項7】 前記主画像処理依頼手段は、処理対象画

像データのうち、詳細な画像データが含まれる部分の複製を生成し、複製した画像データの画像処理要求を出力し、

前記副画像処理依頼手段は、処理が依頼された画像データのうち、詳細な画像データが含まれる部分の複製を生成し、複製した画像データの画像処理要求を出力する、ことを特徴とする請求項5記載の画像処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は画像データを処理する画像処理システムに関し、特に画像データを分散して処理する画像処理システムに関する。

【0002】

【従来の技術】一般に画像データは、文字データや音声データに比べてデータ量が極めて多く、利用者が処理を要求してからその結果を受け取るまでの処理時間が長くなる傾向にある。フィルタリングなどの画像処理の種類によっては利用者が期待する時間以内に画像処理の結果を得ることができない場合もあった。そこで、処理すべき画像データを複数の計算機を使用して分散処理を行うことによる処理の高速化が検討されている。単一の計算機を使用して処理した場合よりも、処理を分散化した場合の方が、利用者が処理を要求してからその結果を受け取るまでの処理時間が短くなるからである。

【0003】例えば、「分散処理環境におけるCGシステムの構築」（情報処理学会第43回全国大会論文集，5U-10，2-517ページ）では、レイトレーシング法などを利用したソフトウェア・レンダラにより高品位なコンピュータ・グラフィックス画像を生成しようとすると、数時間の時間がかかってしまうことが指摘されている。そこで、ネットワーク上に分散した複数台の計算機に処理を最適に分散し、並列処理させることによって、処理に要する時間を短縮しようとしている。

【0004】「クライアント／サーバ型情報処理環境におけるレイトレーシング法による画像生成のモデル化」（情報処理学会第47回全国大会論文集，7E-3，1-223ページ）によれば、レイトレーシング法による作画は画像データの画素個々を独立に計算しているため一つの画像を分割することが容易であることから、クライアント・サーバ・モデルに基づいてレイトレーシング法による作画処理を分散して並列に行う技術についての研究が行われている。つまり、膨大な処理時間を要するレイトレーシング法による画像生成による時間の短縮を、クライアント・サーバ・モデルに基づき、分散し並列処理することによって、短縮することを試みている。

【0005】このような分散処理の実現には、負荷分散の最適化を図る必要がある。そこで、分散処理における負荷分散を最適化するために、一般的にスケジューリングが行われている。そのため、スケジューリング管理を行うためのプロセッサ又はスケジューラなどスケジュー

リングを行うための機構が特別に必要とされていた。ここで、スケジューリングとは各計算機の負荷を最適にすることである。つまり、最も短時間に利用者が期待する画像処理の結果が得られるように、画像処理を分散させることである。なお、一般的な分散処理システムにおいては各計算機における負荷が最適かどうかは、その計算の目的によって様々であり、例えば使用するメモリ量を小さくすることなどもその一つである。また、平均的に負荷を分散させることが最適化の一つの方法である。

【0006】例えば、「情報処理システム」(特開平5-81163号公報)では、通信ネットワーク上のサーバに要求されたサーバ処理をスレッドと呼ばれる実行単位を用いて並列に実行して処理効率を上げようとしている。そのために、このシステムでは、通信ネットワークに分散した複数のコンピュータ上にサービスを要求するクライアント、および複数の種類のサービス要求に対して該当するサービスを選択し実行するサーバを備えている。そして、このサーバにはそれぞれが並列実行することができる四つのスレッドを含んでいる。第1のスレッドはサービス要求の受付を行い、第2のスレッドはサービスを実行し、第3のスレッドはそのサービスの結果の発信(応答)を行い、第4のスレッドは前記三つの各スレッドの実行を制御するとともに、並列実行のためのスケジューリングを行う。

【0007】

【発明が解決しようとする課題】しかし、スケジューリングに要する計算量は膨大であり、その計算には非常に長い時間が必要である。

【0008】例えば、「タスク・ブローカ」という方式がある。この方式では、「アンフィニティ」という選択基準を用いて、処理を行わせるのに最適なマシンを選択する。「アンフィニティ」は、ユーザが直接数値を設定するか、あるいはユーザが記述したプログラムにより生成される。この方式では、最適な計算機を選ぼうとしているが、アプリケーション実行時のCPU負荷及びメモリ使用量並びに計算機のメモリ使用量及びCPU負荷など様々な要素の兼ね合いから値を決定することは容易ではない。

【0009】また、「スケジューリング装置」(特開平5-165397号公報)では、スケジューリング問題の本質は組み合わせ問題であり、考慮すべき資源(ここでは、例えばCPU負荷又はメモリ使用量など)の種類等が増えると問題が複雑になることに着目して、スケジューリングに要する時間の短縮を図っている。即ち、このスケジューリング装置では、冗長制約除去部によって、複数の処理タスク相互の先行制約関係の中から、他の複数のタスク間制約関係の組み合わせ代替表現が可能なタスク間制約関係を冗長なタスク間制約関係として検出し除去する。これにより、先行制約関係を持った複数の処理タスクに資源割当を決定する処理の時間を短縮で

きる。

【0010】このように、スケジューリングを短時間でを行うための様々な方式が考えられているが、依然としてスケジューリングに多くの時間を必要とすることには変わりはない。

【0011】しかも、画像処理によってはある制限時間内又は一定時間内に処理した結果を必要とする処理があり、この場合には、所定の時間以内に必ず画像処理を終了させなければならない。例えば、リアルタイム・アニメーションはその一つの例である。つまり、リアルタイム・アニメーションにおいては均一なフレーム速度で画像を表示することが必要である。一般に静止画像を単純に連続して生成するリアルタイム・アニメーション・システムでは、光景(シーン)の複雑さによってフレーム速度が変動してしまう。そこで、リアルタイム・アニメーションでは、極力画質は高く維持し、さらに指定された目標フレーム時間内に各フレームの静止画像を生成する必要がある。

【0012】本発明はこのような点に鑑みてなされたものであり、スケジューリングのような複雑な処理を行わずに、画像データの分散処理を行う画像処理システムを提供することを目的とする。

【0013】また、本発明の他の目的は、利用者が所望する時間内に処理した画像の結果を戻すことを可能とする画像処理システムを提供することである。

【0014】

【課題を解決するための手段】本発明では上記課題を解決するために、画像データを分散して処理する画像処理システムにおいて、入力された処理対象画像データの一部を複製し、複製した画像データの画像処理要求を出力する主画像処理依頼手段と、前記処理対象画像データに所定の処理を行うとともに、処理済画像データを受け取ると、受け取った処理済画像データと自己の処理した画像データとを合成し、所定のタイミングで画像処理結果データとして出力する主画像処理実行手段と、を有する主画像処理装置と、画像処理要求を受け取ると、処理が依頼された画像データの一部を複製し、複製した画像データの画像処理要求を出力する副画像処理依頼手段と、処理が依頼された画像データに対して所定の処理を行い、処理済画像データとして前記主画像処理実行手段に転送する副画像処理実行手段と、を有する複数の副画像処理装置と、を有することを特徴とする画像処理システムが提供される。

【0015】このような画像処理システムによれば、処理対象画像データが主画像処理装置に入力されると、主画像処理依頼手段により、処理対象画像データの一部が複製され、画像処理要求が出力される。その画像処理要求を副画像処理装置が受け取ると、副画像処理依頼手段により、処理が依頼された画像データの一部が複製され、画像処理要求が出力される。そして、主画像処理実

行手段は処理対象画像データの処理を行い、副画像処理実行手段は処理が依頼された画像データの処理を行う。副画像処理実行手段が処理した処理済画像データは主画像処理実行手段に転送され、主画像処理実行手段の処理した画像データと合成され、その後、所定のタイミングで画像処理結果データとして出力される。

【0016】これにより、スケジューリング管理のための機構を用いることなく画像データの分散処理を行うことができる。

【0017】

【発明の実施の形態】以下、本発明の実施の形態を図面を参照して説明する。図1は本発明の原理構成図である。本発明の画像処理システムは、一つの主画像処理装置1と、複数の副画像処理装置2、3とで構成されている。

【0018】主画像処理装置1内には、主画像処理依頼手段1aと主画像処理実行手段1bとが設けられている。主画像処理依頼手段1aは、処理対象画像データ4が入力されると、処理対象画像データ4の一部を複製し、それを画像処理要求とともに出力する。主画像処理実行手段1bは、処理対象画像データ4の処理を実行するとともに、副画像処理装置からの処理済画像データを受け取ると、自己の処理した画像データと合成する。そして、主画像処理実行手段1bは、全ての処理対象画像データ4の処理が終了するか、あるいは所定の制限時間に達したら、画像処理結果データ5を出力する。

【0019】副画像処理装置2、3には、それぞれ副画像処理依頼手段2a、3aと副画像処理実行手段2b、3bとが設けられている。副画像処理依頼手段2a、3aは、画像処理要求を受け取る。画像処理要求を受け取った副画像処理依頼手段2a、3aは、画像処理要求とともに受け取った画像データの一部を複製し、それを画像処理要求とともに出力する。副画像処理実行手段2b、3bは、画像処理要求とともに受け取った画像データの処理を実行する。そして、副画像処理実行手段2b、3bは、画像データの処理が終了するか、あるいは所定の制限時間に達したら、処理後の画像データを主画像処理装置1へ送信する。

【0020】このような画像処理システムによれば、ユーザが、処理対象画像データ4を主画像処理装置1に入力すると、主画像処理依頼手段1aにより、処理対象画像データの一部が副画像処理装置2、3のいずれかに渡される。画像データを受け取った副画像処理装置内の副画像処理依頼手段は、さらに画像データの一部を複製し、それを他の副画像処理装置に渡す。これにより、画像データは、次々と複製される。そして、主画像処理装置1と副画像処理装置2、3は、自己が受け取った画像データを処理する。そして、処理が終了するか、あるいは画像処理開始から所定の時間が経過すると、処理済画像データが副画像処理装置2、3から主画像処理装置1

に渡される。主画像処理装置1の主画像処理実行手段1bは、副画像処理装置2、3から受け取った処理済画像データと、自己が処理した画像データとを合成し、画像処理結果データ5として出力する。

【0021】このようにして、スケジューリングを行わずに、複数の装置で分散して画像処理を行うことができる。しかも、画像処理結果データ5は、常に所定の時間内に出力される。

【0022】ところで、以上のような画像処理システムは、ネットワークを介して接続された複数の計算機により実現することができる。そこで、LAN（ローカル・エリア・ネットワーク）を介して接続された複数台の計算機による本発明を実施の形態を以下に説明する。

【0023】図2は本発明の第1の実施の形態の概略構成を示す図である。図に示すように、ネットワーク11を介して複数の計算機100、200、300が接続されている。ここで、計算機100内に生成されるプロセスが図1に示す主画像処理装置1となり、計算機200、300内に生成されるプロセスが図1に示す副画像処理装置2、3となる。

【0024】そして、ネットワーク11によって接続された複数の計算機において生成されるプロセスを、処理すべき画像処理の実行を担うという意味でサーバと呼ぶことにする。各サーバには各々を識別するために識別番号が付けられる。これをサーバIDと呼ぶことにする。サーバIDはネットワーク11によって接続された複数の計算機において一意に決められ、0から始まるものとする。各サーバにサーバIDを付加して、例えば、サーバ0号、サーバn号などと呼ぶことにする。ここで、nはサーバIDである。

【0025】さらに、サーバの計算機上の生存時間、つまりサーバが生成されてから消滅するまでの時間をライフ・タイムと呼ぶことにする。図3はサーバの内部構成を示すブロック図である。各計算機100、200には、サーバ0号110、サーバn号210が生成される。なお、サーバIDの付け方の規則については、後述する。また、他の計算機300に生成されるサーバの構成は、サーバn号210と同様の構成である。

【0026】サーバ0号110は、制御部111、利用者インタフェース部112、時間管理部113、画像処理部115、サーバ生成部114、画像処理結果受付部116から構成される。制御部111は、利用者インタフェース部112、時間管理部113、画像処理部115、サーバ生成部114及び画像処理結果受付部116の動作を制御する。利用者インタフェース部112は利用者12との画像データの受け渡しなどの処理を行う。利用者12は利用者インタフェース部112を通じてサーバ0号110と画像データの受け渡しを行う。時間管理部113はサーバ0号110の存在する計算機100の時計（システム・クロック）の時刻、または処理時間

を管理する。画像処理部115は画像データを記憶し、フィルタリングなどの画像処理を行う。サーバ生成部114は他の異なる計算機に対してサーバの生成を行う。このサーバの生成は、RPC（リモート・プロシジャ・コール）により行われる。画像処理結果受付部116はサーバn号210から戻された画像処理結果を受け付ける。

【0027】サーバn号210は、制御部211、時間管理部212、画像処理部213、サーバ生成部214、及び画像処理結果送出处215から構成される。制御部211は、時間管理部212、画像処理部213、サーバ生成部214、及び画像処理結果送出处215の動作を制御する。時間管理部212はサーバn号210の存在する計算機200の時計（システム・クロック）の時刻、または処理時間を管理する。画像処理部213は画像処理部115と同様に画像データを記憶し、フィルタリングなどの画像処理を行う。サーバ生成部214は他の異なる計算機に対してサーバの生成を行う。画像処理結果送出处215は画像処理部213で処理した結果をサーバ0号110に戻す処理を行う。

【0028】なお、前述のサーバ0号110の処理手続きにおける処理開始時刻 t_s 及びライフ・タイム t_l の設定並びに現在時刻 t_c の取得は時間管理部113が行う。また、サーバn号210における現在時刻 t_c の取得は時間管理部212が行う。サーバ0号110の画像処理結果受付部116は画像処理結果データ受付手続きにおいて利用されるキューを持つ。

【0029】以上のような構成の画像処理システムにおいて、以下のような手順で画像処理を行う。図4は第1の実施の形態のサーバ0号110の処理手順を示すフローチャートである。この処理は、利用者12がフィルタリング処理など画像処理をサーバ0号110に依頼することにより開始される。以下、処理手順をステップに沿って説明する。

〔S11〕利用者インタフェース部112は、利用者12からの画像処理要求を受け付ける。このとき、利用者12は処理すべき画像データを指定している。

〔S12〕時間管理部113は、処理開始時刻 t_s 及び最大ライフ・タイム t_l を設定する。ここで、最大ライフ・タイム t_l とは、利用者12が画像処理のために待つことのできる時間である。処理開始時刻 t_s と最大ライフ・タイム t_l とが設定された時間管理部113は、処理開始時刻を t_s とし、処理終了時刻、すなわちサーバ0号110が利用者12に画像を処理した結果を戻す時刻を t_e とする。また、以後処理開始時刻 t_s から処理終了時刻 t_e までの時間を t_{es} と表記し（ $t_{es} = t_e - t_s$ ）、処理開始時刻 t_s から現在の時刻 t_c までの経過時間を t_{cs} と表記する（ $t_{cs} = t_c - t_s$ ）。

【0030】サーバ0号110の実行時のライフ・タイ

ム（実際の結果的なライフ・タイム）は t_{es} であり、 t_{es} は t_l 以下（ $t_l \geq t_{es} > 0$ ）でなければならない。よって、サーバ0号110のライフ・タイムは処理時間を保証するものとなる。なお、 t_l は利用者12自身がサーバ0号110に与えてもよいし、前述した如くサーバ0号110自身が規定値を設定してもよい。

〔S13〕サーバ生成部114は、処理開始時刻 t_s に直ちに自分と同じ処理をするサーバを他の二つの異なる計算機にそれぞれ生成することを試みる。ただし、そのサーバ生成において、処理すべき画像データの一部しか複製しない。ここで、生成した偶数IDのサーバに対して処理すべきデータの前半部分を複製し、生成した奇数IDのサーバに対して処理すべきデータの後半部分を複製する。なお、サーバ生成に失敗した場合の特別な処理はない。

〔S14〕時間管理部113は、現在の時刻 t_c を取得する。

〔S15〕時間管理部113は、経過時間 t_{cs} （ $= t_c - t_s$ ）と最大ライフ・タイム t_l とを比較する。経過時間 t_{cs} が最大ライフ・タイム t_l に達していなければステップS16に進み、達していればステップS18に進む。

〔S16〕生存時間あり、すなわち $t_l \geq t_{cs}$ の場合、画像処理部115は、サーバn号から処理した結果のデータが戻されるのを待つ。ここで、サーバn号は0以外のサーバIDを持つサーバである。この手続きについてはサーバ0号の画像処理結果データ受付手続きとして後述する。画像処理部115は、その待ち時間を利用して、処理すべき画像データに対する画像処理を画像データの先頭から処理すべきデータを最小の実行単位分だけ実行する。

【0031】なお、サーバn号から処理した結果のデータが戻されたとき、画像処理部115は、その結果のデータを自己の画像データの一部として複製し、その部分は処理済、つまり処理しないこととする。ここで、最小の実行単位とは、例えば、スキャンライン、ブロック、あるいは画素などの単位である。

〔S17〕画像処理部115は、処理すべきデータの有無を判断する。処理すべきデータがある場合、ステップS15に戻る。一方、処理すべきデータはない場合、ステップS18に進む。

〔S18〕ステップS15において生存時間なしと判断された場合、またはステップS17において処理すべきデータがないと判断された場合、画像処理部115は、利用者インタフェース部112を介して、直ちに自己の持つ画像処理結果データを利用者に戻す。

【0032】図5は第1の実施の形態のサーバn号（ここで、nは0以外のサーバIDである）210の処理手順を示すフローチャートである。この処理は、サーバ0号110からの画像処理要求を、サーバn号210が受

け取った際に開始される。

〔S21〕サーバ生成部214は、直ちに自分と同じ処理をするサーバを他の二つの異なる計算機にそれぞれ生成を試みる。ただし、そのサーバ生成において、処理すべき画像データの一部しか複製しない。ここで、生成した偶数IDのサーバに対して処理すべきデータの前半部分を複製し、生成した奇数IDのサーバに対して処理すべきデータの後半部分を複製する。なお、サーバ生成に失敗した場合の特別な処理はない。

〔S22〕画像処理部213は、処理すべき画像データに対する画像処理を画像データの先頭から実行する。

〔S23〕時間管理部212は、現在の時刻 t_c を取得する。

〔S24〕時間管理部212は、経過時間 t_{cs} ($=t_c - t_s$) と最大ライフ・タイム t_l とを比較する。経過時間 t_{cs} が最大ライフ・タイム t_l に達していなければステップS25に進み、達していれば制御部211が自己のプロセスを消滅させ処理を終了する。

〔S25〕生存時間あり、すなわち $t_l \geq t_{cs}$ の場合、画像処理結果送出部215は、直ちに自己の持つ画像処理結果データをサーバ0号110に戻す。そして、制御部211は自己のプロセスを消滅させる。

【0033】以上の様な処理を行うサーバが各計算機内に生成されることより、画像データが、各サーバで分散処理される。図6は第1の実施の形態におけるサーバIDの付け方の規則を示す図である。なお、サーバIDの表記された長方形は各サーバが処理すべきデータであることを示す。サーバが生成されるたびに生成されたサーバで処理されるべき画像データは小さくなる。つまり、二つのサーバを生成したサーバは画像データの前半部分を一つのサーバに複製し、その画像データの後半部分を他のサーバに複製する。ここでは、サーバが第4世代まで生成されていることを示している。

【0034】サーバ0号31から生成されるサーバをサーバ00号32a及びサーバ01号32bとする。サーバ00号32aはサーバ000号33a及びサーバ001号33bを生成する。また、サーバ01号32bはサーバ010号33c及びサーバ011号33dを生成する。以下同様に、サーバ000号34a、サーバ0001号34b、サーバ0010号34c、サーバ0011号34d、サーバ0100号34e、サーバ0101号34f、サーバ0110号34g、及びサーバ0111号34hが生成される。つまり、サーバは自己のサーバIDに0又は1を右から付加した番号をサーバIDとするサーバを二つ生成する。これをサーバを生成することに繰り返す。

【0035】図7はサーバ001号がサーバの生成に失敗した状況を示す図である。図6と同様に、サーバ0号31から順にサーバが生成されるが、サーバ001号33bがサーバの生成に失敗すると、図6に示したサーバ

0010号34cとサーバ0011号34dとは生成されない。

【0036】ところで、あるサーバが別のサーバを生成する場合のサーバを生成する計算機の決定には、ブロードキャストを用いることができる。即ち、サーバ生成部は、先ずサーバを生成したい旨の画像処理要求をネットワーク上の全ての計算機に送信する。その要求を受け取った計算機は、自己にサーバを生成できるか否かを判断する。判断基準としては、例えば、1台の計算機では一つのサーバしか作らないものとし、既にサーバが生成されていればそれ以上生成しないものとする。また、他の情報処理の為に計算機が使用されておりその負荷が大きい場合にも、その計算機はサーバを生成しないものとする。そして、サーバが生成可能であると判断した計算機のみ、要求を出力したサーバに返事を送り返す。

【0037】返事を受け取ったサーバは、最初に返事を返した計算機に対して、複製した画像データを送信する。この画像データを受け取った計算機は、内部にサーバn号を生成し画像処理を行う。

【0038】次に、サーバ0号におけるサーバn号からの画像処理結果の受付の手続きについて説明する。一般的な分散処理環境、つまりネットワークで接続された複数の計算機において画像処理を行う環境においては、各計算機的能力が異なることが予想されるので、各サーバn号から部分的な画像処理結果が任意の時刻に到達する場合がある。このような場合についての処理手順を以下に示す。

【0039】図8はサーバ0号の画像処理結果データ受付状況を示す図である。この図は、サーバ001号33b及びサーバ0010号34cがそれぞれ各サーバで処理した結果をサーバ0号31に戻す様子を示している。サーバ001号33b又はサーバ0010号34cなどサーバn号がサーバ0号31に各サーバで処理した結果を戻すタイミングによっては、その戻された画像処理結果データとサーバ0号31で処理した結果が重複する可能性がある。この図では、ハッチ部分が処理された画像データである。

【0040】図9は画像処理結果データが重複した場合におけるタイミング・チャートである。サーバ0号31は、時刻 t_s で利用者から受け取った画像データの処理を開始する。生成されたサーバ001号33b及びサーバ0010号34cは時刻 t_c において、それぞれのサーバで処理した結果をサーバ0号31に戻す。

【0041】ここで、サーバ0号31における各サーバからの画像処理結果データの受付処理については、キュー、すなわち受付データ用の先入れ先出し型のバッファ記憶装置を用いる方法による手続きが考えられる。これは、サーバ001号33bとサーバ0010号34cが同じ時刻 t_c に処理した結果を戻した場合にも、それらの画像処理結果データが前述のキューに配置されると、

それらの先後関係が生じることを利用するものである。なお、画像処理結果データのバッファ記憶装置への格納は、サーバ内の画像処理結果受付部が行う。

【0042】図10は画像処理結果データを受け付けたサーバ0号の処理手順を示す図である。この処理は、全てサーバ0号内の画像処理部が行う。

〔S31〕受付データ用の先入れ先出し型バッファ記憶装置における画像処理結果データの有無を検査する。画像処理結果データがある場合にはステップS32に進み、画像処理結果データがない場合には処理を終了する。

〔S32〕画像処理結果データがある場合、受け付けたデータについて利用者から受け取った時点での配置情報（アドレスあるいは座標）を検査する。

〔S33〕ステップS32で得た配置情報から、バッファ記憶装置内の画像処理結果データが処理済のデータであるかどうかを判断する。処理済のデータであれば、この処理を終了する。処理済でないデータが含まれていればステップS34に進む。

〔S34〕処理済であるデータ以外の部分（未処理部分のデータ）を自己（サーバ0号）のデータとして複製し、その部分は処理済とする。

【0043】以上のようにして、スケジューリングを行わずに処理すべき画像データを複数のサーバで分散処理することができる。しかも、時間管理部が最大ライフ・タイム t_l を管理しているため、処理途中であっても、一定の時間以内に画像処理結果データを利用者に返すことができる。

【0044】次に第2の実施の形態について説明する。第1の実施の形態では、サーバを二つずつ生成する場合の例を示したが、生成するサーバの数は一つであってもよいし、あるいは三つ以上であってもよい。そこで、サーバを一つずつ生成する場合を、第2の実施の形態として以下に説明する。なお、各計算機の接続関係と各サーバ内の基本的な構成は、図2、図3で示した第1の実施の形態のものと同様である。そこで、第2の実施の形態における各構成要素を図2、図3で示したものと同じ名称で呼ぶこととして、第2の実施の形態の処理内容を説明する（以後の他の実施の形態においても同様）。

【0045】図11は第2の実施の形態におけるサーバ0号の処理手順を示すフローチャートである。なお、サーバIDの付け方の規則については、後述する。

〔S41〕利用者インタフェース部は、利用者からの画像処理要求を受け付ける。このとき、利用者は処理すべき画像データを指定している。

〔S42〕時間管理部は、処理開始時刻 t_s 及び最大ライフ・タイム t_l を設定する。

〔S43〕サーバ生成部は、直ちに自分と同じ処理をするサーバを他の一つの異なる計算機に生成するための処理を行い、その成否を判断する。サーバの生成が失敗す

ればステップS44に進み、成功すればこのステップS43を繰り返す。従って、サーバが生成できなくなるまで、サーバの生成処理を繰り返すことになる。

【0046】サーバの生成に成功した場合、処理すべき画像データの一部しか複製しない。ここで、サーバ0号は、処理すべきデータの前半部分を自己の処理すべきデータとし、処理すべきデータの後半部分を生成したサーバに生成したサーバの処理すべきデータとして複製する。なお、利用者がサーバ0号に与えた処理すべき画像データと、サーバ0号の処理手続きが進むごとに小さくなっていくサーバ0号が処理すべき画像データとの差分（余りの部分）、すなわちサーバ n 号が本来処理すべき部分を処理可能なデータと呼ぶことにする。

〔S44〕時間管理部は、現在の時刻を取得する。

〔S45〕時間管理部は、経過時間 t_{cs} と最大ライフ・タイム t_l とを比較する。生存時間あり、すなわち $t_l \geq t_{cs}$ の場合ステップS46に進み、生存時間なし、すなわち $t_l < t_{cs}$ の場合ステップS48に進む。

〔S46〕画像処理部は、サーバ n 号から処理した結果のデータが戻されるのを待つ。その待ち時間を利用して、処理すべき画像データ又は処理可能なデータに対する画像処理を画像データの先頭から処理すべきデータを最小の実行単位分だけ実行する。

〔S47〕画像処理部は、処理すべきデータ及び処理可能なデータの有無を判断する。処理すべきデータ又は処理可能なデータが有る場合、ステップS45に戻り、処理すべきデータ及び処理可能なデータが無い場合、ステップS48に進む。

〔S48〕ステップS45において生存時間なしと判断された場合、またはステップS47において処理すべきデータがなく、かつ処理可能なデータがないと判断された場合、画像処理部は、利用者インタフェース部を介して、直ちに自己の持つ画像処理結果データを利用者に返す。

【0047】サーバ n 号の処理手続き（ここで、 n は0以外のサーバIDである。）は第1の実施の形態と同じであるで説明を省略する。ただし、サーバの生成処理（図5のステップS21に該当）において処理すべき画像データの一部しか複製しない点が異なる。ここで、サーバ n 号は、処理すべきデータの前半部分を自己の処理すべきデータとし、処理すべきデータの後半部分を生成したサーバ（サーバ $n+1$ 号）に生成したサーバの処理すべきデータとして複製する。

【0048】さらに、サーバ n 号のよって作られたサーバは、別のサーバの生成は行わない。即ち、サーバ0号によって作られたサーバ n 号のみが別のサーバを生成することができる。

【0049】図12は第2の実施の形態におけるサーバIDの付け方を示す図である。サーバ0号41から生成

されるサーバはサーバ01号44とする。サーバ01号44を生成したサーバ0号41はそのサーバIDを00に更新する。すなわち、サーバ0号41はサーバ00号42となる。サーバ00号42は、サーバ001号47を生成し、サーバ000号43となる。また、サーバ01号44は、サーバ011号46を生成し、サーバ010号45となる。つまり、生成されたサーバは元のサーバIDに1を右から付加した番号がサーバIDとなる。一方、サーバを生成したサーバは自己のサーバIDに0を右から付加してサーバIDを更新する。これをサーバを生成するごとに繰り返す。ここで、サーバ00号42、サーバ000号43など二進記法の読み方で識別番号が0となるものはサーバ0号であるとみなす。

【0050】図13は第2の実施の形態における各サーバ0号の処理すべき画像データ及び処理可能データの両領域の変化を示す図である。この図において、斜線のハッチングの部分はサーバ0号が処理すべき画像データを示しており、クロスハッチの部分はサーバ0号が処理可能な画像データを示している。サーバ0号41が処理を開始する時点では、サーバ0号41において利用者から受け取った画像データのすべてが処理すべき画像データとなっている。この時点ではサーバn号、つまりサーバ0号41以外のサーバは存在せず、サーバn号が処理すべき画像データはない。次に、サーバ0号41がサーバ01号を生成し、自らがサーバ00号42となった時点で、サーバ01号に部分的に複製した画像データの部分をサーバ00号42が処理可能な画像データとする。他の部分がサーバ00号42が処理すべき部分となる。同様にサーバが生成されるごとにサーバ0号はサーバ00号43、サーバ000号44となると、処理すべき画像データの領域は小さくなり、処理可能な画像データの領域は大きくなる。

【0051】次に第3の実施の形態について説明する。上記の第2の実施の形態は他の異なる計算機に一つのサーバを生成して画像を処理する場合であった。しかしサーバを生成したサーバ0号のIDが更新され、さらにそのサーバ0号は他の異なる計算機に一つのサーバを生成する。このとき、直前にサーバ0号により生成されたサーバn号も他の異なる計算機に一つのサーバを生成しているため、結局第1の実施の形態と同様に概念的にはサーバを二つずつ生成することを意味していた。そこで、全てのサーバが他の異なる計算機に一つのサーバを生成して画像を処理する場合、つまり実質的に一つずつしかサーバを生成しない場合の例を第3の実施の形態として以下に説明する。

【0052】第3の実施の形態におけるサーバ0号、及びサーバn号の処理手順は第1の実施の形態の処理手順(図4、図5に示す)とほぼ同じである。そこで、第1の実施の形態における相違点のみを、図4、図5に示すステップ番号を引用して説明する。なお、第3の実施の

形態におけるサーバIDの付け方の規則については、後述する。

【0053】第3の実施の形態のサーバ0号の処理手続きにおける第1の実施の形態との相違点は、図4のステップS13における処理である。これは、サーバ0号が、直ちに自分と同じ処理をするサーバ1号を他の計算機に生成する処理である。この処理において、第3の実施の形態では、他のサーバを一つしか生成せず、その生成されたサーバには処理すべき画像データの一部しか複製しない。ここで、サーバ0号は、処理すべきデータの前半部分を自己の処理すべきデータとし、処理すべきデータの後半部分を生成したサーバ1号にサーバ1号の処理すべきデータとして複製する。

【0054】第3の実施の形態のサーバn号の処理手続きにおける第1の実施の形態との相違点は、図5のステップS21における処理である。これは、サーバn号が、直ちに自分と同じ処理をするサーバn+1号を他の計算機に生成する処理である。この処理において、第3の実施の形態では、他のサーバを一つしか生成せず、その生成されたサーバには処理すべき画像データの一部しか複製しない。ここで、サーバn号は、処理すべきデータの前半部分を自己の処理すべきデータとし、処理すべきデータの後半部分を生成したサーバn+1号に処理すべきデータとして複製する。なお、サーバ生成に失敗した場合の特別な処理はなく、次の手続きに進む。

【0055】図14は第3の実施の形態におけるサーバIDの付け方の規則を示す図である。サーバ0号51から生成されるサーバはサーバ1号52とする。サーバ1号52はサーバ2号53を生成する。また、サーバ2号53はサーバ3号54を生成する。つまり、生成されたサーバは元のサーバIDに1を加えた番号がサーバIDとなる。これをサーバを生成するごとに繰り返す。

【0056】このようにしてサーバを生成していけば、新たなサーバは一つずつしか生成されない。第2の実施の形態では、サーバ0号はサーバの生成に失敗するまでサーバn号を作り続けるため、サーバの生成に失敗しなければ画像処理を行うことができない。ところが、第3の実施の形態では、サーバ0号は1回しかサーバの生成処理を行わないため、サーバの生成処理を1回だけ行った後、直ぐに画像処理を行うことができる。

【0057】次に第4の実施の形態について説明する。第1の実施の形態、第2の実施の形態及び第3の実施の形態においては、各計算機におけるサーバの終了(消滅)を単一のライフ・タイムt1のみによって制御する場合の手続きを示した。ここでは、各計算機におけるサーバの当該計算機上での生存時間を各々制御する場合を、第4の実施の形態としてその手続きを示す。

【0058】ここで、各サーバの計算機上での生存時間を決定するための時間をデッドラインと呼ぶことにし、サーバIDとともにtd(0)などと記することに

する。つまり、サーバ0号のデッドライン $t_d(0)$ はライフ・タイム t_l と同一である。サーバ1号のデッドラインは $t_d(1)$ と示すことにする。

【0059】以下に先の第3の実施の形態における各サーバの生存時間制御方式を変更し、かつ、サーバ0号以外のサーバ、つまりサーバ n 号の処理手順を一部変更した場合の手続きについて示す。なお、サーバ1Dの付け方は、第3の実施の形態と同様とする。

【0060】図15は第4の実施の形態におけるサーバ0号の処理手順を示すフローチャートである。

〔S51〕利用者インタフェース部は、利用者からの画像処理要求を受け付ける。このとき、利用者は処理すべき画像データを指定している。

〔S52〕時間管理部は、処理開始時刻 t_s 及び最大ライフ・タイム t_l を設定する。ここで、最大ライフ・タイム t_l とは、利用者12が画像処理のために待つことのできる時間である。処理開始時刻 t_s と最大ライフ・タイム t_l が設定された時間管理部は、処理開始時刻を t_s とし、処理終了時刻、すなわちサーバ0号が利用者に画像を処理した結果を戻す時刻を t_e とする。また、以後処理開始時刻 t_s から処理終了時刻 t_e までの時間を t_{es} と表記し($t_{es}=t_e-t_s$)、処理開始時刻 t_s から現在の時刻 t_c までの経過時間を t_{cs} と表記する($t_{cs}=t_c-t_s$)。

【0061】サーバ0号の実行時のライフ・タイム(実際の結果的なライフ・タイム)は t_{es} であり、 t_{es} は t_l 以下($t_l \geq t_{es} > 0$)でなければならない。よって、サーバ0号のライフ・タイムは処理時間を保証するものとなる。なお、 t_l は利用者自身がサーバ0号に与えてもよいし、サーバ0号自身が規定値を設定してもよい。また、ライフ・タイム t_l はサーバ0号のデッドライン $t_d(0)$ と同じ($t_l=t_d(0)$)である。

〔S53〕サーバ生成部は、直ちに自分と同じ処理をするサーバ1号を他の計算機に生成する。ただし、その生成において処理すべき画像データの一部しか複製しない。ここで、サーバ0号は、処理すべきデータの前半部分を自己の処理すべきデータとし、処理すべきデータの後半部分を生成したサーバ1号にサーバ1号の処理すべきデータとして複製する。また、生成されたサーバ1号のデッドライン $t_d(1)$ は、サーバ0号のデッドライン $t_d(0)$ の半分であるように設定する。つまり、 $t_d(1)=t_d(0)/2$ とする。なお、サーバ生成に失敗した場合の特別な処理はない。

〔S54〕時間管理部は、現在の時刻 t_c を取得する。

〔S55〕時間管理部は、サーバ0号は、経過時間 $t_{cs}(=t_c-t_s)$ とデッドライン $t_d(0)$ とを比較する。経過時間 t_{cs} がデッドライン $t_d(0)$ に達していなければステップS56に進み、達していればステップS58に進む。

〔S56〕生存時間あり、すなわち $t_d(0) \geq t_{cs}$ の場合、画像処理部は、サーバ n 号から処理した結果のデータが戻されるのを待つ。そして、その待ち時間を利用して、処理すべき画像データに対する画像処理を画像データの先頭から処理すべきデータを最小の実行単位分だけ実行する。

【0062】なお、サーバ n 号から処理した結果のデータが戻されたとき、その結果のデータを自己の画像データの一部として複製し、その部分は処理済、つまり処理しないこととする。ここで、最小の実行単位とは、例えば、スキャンライン、ブロック、あるいは画素などの単位である。

〔S57〕画像処理部は、処理すべきデータの有無を判断する。処理すべきデータがある場合、ステップS55に戻る。一方、処理すべきデータはない場合、ステップS58に進む。

〔S58〕ステップS55において生存時間なしと判断された場合、またはステップS57において処理すべきデータがないと判断された場合、画像処理部は、利用者インタフェース部を介して、直ちに自己の持つ画像処理結果データを利用者に戻す。

【0063】図16は第4の実施の形態におけるサーバ n 号の処理手順を示すフローチャートである。

〔S61〕サーバ生成部は、直ちに自分と同じ処理をするサーバ $n+1$ 号を他の計算機に生成する。ただし、その生成において処理すべき画像データの一部しか複製しない。ここで、サーバ n 号は、処理すべきデータの前半部分を自己の処理すべきデータとし、処理すべきデータの後半部分を生成したサーバ $n+1$ 号に処理すべきデータとして複製する。また、生成されたサーバ $n+1$ 号のデッドライン $t_d(n+1)$ は、サーバ n 号のデッドライン $t_d(n)$ の半分であるように設定する。つまり、 $t_d(n+1)=t_d(n)/2$ とする。

〔S62〕時間管理部は、現在の時刻 t_c を取得する。

〔S63〕時間管理部は、経過時間 $t_{cs}(=t_c-t_s)$ とデッドライン $t_d(n)$ とを比較する。経過時間 t_{cs} が $t_d(n)$ 以下ならばステップS64に進み、 $t_d(n)$ を超えていれば制御部は自己のプロセスを消滅させる。

〔S64〕生存時間あり、すなわち $t_d(n) \geq t_{cs}$ の場合、画像処理部は、処理すべき画像データに対する画像処理を画像データの先頭から実行する。

〔S65〕画像処理結果送出部は、直ちに自己の持つ画像処理結果データをサーバ0号に戻す。そして、制御部は自己のプロセスを消滅させる。

【0064】ここでは、生成されたサーバ $n+1$ 号のデッドライン $t_d(n+1)$ は、サーバ n 号のデッドライン $t_d(n)$ の半分であるように設定したが、他の規則によって設定してもよい。つまり、生成されたサーバ $n+1$ 号のデッドライン $t_d(n+1)$ が、サーバ n 号の

デッドライン $t_d(n)$ の3分の1であるように設定してもよい。

【0065】ここで、各サーバの動作と処理に要する時間の関係について説明する。図17及び図18は各サーバが生成されるタイミングと当該サーバで画像データが処理される時間について示している。

【0066】図17は理想的な状況における各サーバが生成されるタイミングと当該サーバで画像データが処理される時間示す図である。つまり、サーバの生成及びデータの複製に時間を要しない場合、あるいはサーバの生成及びデータの複製に時間が無限に小さい場合のようすを示している。また、ネットワークには4台の計算機が接続され、その計算機の能力が等しいことが仮定されている。

【0067】このような場合において前述した処理手順に従って、まずライフ・タイム t_l が設定される。このライフ・タイム t_l とサーバ0号61のデッドラインは等しい。また、図17においては利用者から依頼された処理すべき画像を処理するのに必要十分なライフ・タイム t_l が設定されている場合を示している。つまり、利用者から依頼された処理すべき画像を処理するのに要する時間をサーバ0号61のサーバIDでラベルづけされた四角形の枠で示している。その四角形の横幅とライフ・タイム t_l の長さは等しい。ここで、サーバ0号61は直ちにサーバ1号62を生成する。サーバ0号61の処理すべきデータを示すハッチ部分、すなわち利用者から指定された画像データの前半部分をサーバ0号61が処理する。それ以外の部分（ハッチのかかっていない部分）はサーバ1号62が処理できるように複製される。なお、この部分はサーバ0号61が処理可能な部分である。また、サーバ1号62のデッドライン $t_d(1)$ は $t_d(0)$ の2分の1と等しいように、つまりライフ・タイム t_l の半分であるように設定される。

【0068】サーバ0号61から生成されたサーバ1号62は直ちにサーバ2号63を生成することを試みる。サーバ1号62から生成されたサーバ2号63は直ちにサーバ3号64を生成することを試みる。さらに、サーバ2号63から生成されたサーバ3号64は直ちにサーバ4号を生成することを試みる。

【0069】図17においては、ネットワークに4台の計算機しか接続されていないため、サーバ1号62、サーバ2号63及びサーバ3号64の生成には成功したが、サーバ4号の生成には失敗した場合を示している。また、サーバ2号63のデッドラインはサーバ1号62のその半分に、サーバ3号64のデッドラインはサーバ2号63のその半分にそれぞれ設定されている。つまり、 $t_d(2)$ は $t_d(1)$ の2分の1であり、 $t_d(3)$ は $t_d(2)$ の2分の1である。

【0070】さらに、サーバ2号63の処理すべき画像データは、サーバ1号62に対してサーバ0号61から

複製された画像データの後半部分である。同様に、サーバ3号64の処理すべき画像データは、サーバ2号63に対してサーバ1号62から複製された画像データの後半部分である。

【0071】各サーバは自己の処理すべき画像データ、すなわち図17において各サーバが処理すべき画像データであることを示すハッチ部分に相当する画像データを処理する。サーバ2号63及びサーバ3号64は処理開始時刻 t_s からそれぞれの処理すべき画像データを処理し、同じ時刻で処理が終了している。つまり、以後サーバIDを n とし、各サーバの処理終了時刻を $t_e(n)$ と表記することになると、時刻 $t_e(2)$ でサーバ2号63が処理を終了し、その同じ時刻 $t_e(3)$ でサーバ3号64が処理が終了している。すなわち、 $t_e(2)$ と $t_e(3)$ は等しい ($t_e(2) = t_e(3)$)。 $t_e(2)$ 又は $t_e(3)$ の時点ではサーバ2号63及びサーバ3号64はそれぞれ各サーバで処理した画像処理の結果をサーバ0号61に戻している。また、時刻 t

(1) の時点ではサーバ1号62はその結果をサーバ0号61に戻し処理を終了している。サーバ0号61は時刻 $t(0)$ の時点では、すでに自己の処理すべき画像データの処理を終了し、またサーバ1号62、サーバ2号63及びサーバ3号64からそれぞれ処理した結果を受け取り、さらにすべての部分の画像データを利用者に戻している。なお、この場合サーバ3号64の処理開始時刻 t_s から処理終了時刻 $t_e(3)$ までの時間とサーバ3号64のデッドライン $t_d(3)$ は等しい。また、サーバ0号61はライフ・タイム t_l の半分の時間で利用者に画像を処理した結果を戻している。

【0072】図18は実際の各サーバが生成されるタイミングと当該サーバで画像データが処理される時間を示す図である。つまり、図17においてはサーバの生成及びデータの複製に時間を要しない場合、あるいはサーバの生成及びデータの複製に時間が無限に小さい場合のようすを示しているが、図18はサーバの生成及びデータの複製にある程度の時間を要した場合について、それらの処理に要した時間などを示している。すなわち、図18において、各サーバの処理すべき画像データを示す領域71a、72a、73a、74aの左右に隣接している領域81～86は、サーバの生成及び画像データの部分複製などの処理を示す。図18は、サーバの生成及び画像データの部分複製に要する時間がフィルタリングなどの実質的な画像処理に要する時間の8分の1である場合を示している。つまり、各サーバの処理すべき画像データを処理するのに要する時間を示す領域71a、72a、73a、74aの横幅 T_x は、その領域の直ぐ左側に隣接した、各サーバの生成及び画像データの部分複製に要する時間を示す領域81～83の横幅 T_y の8倍である。

【0073】サーバ0号71は、サーバ1号72を生成

し画像データを部分複製するのに領域81の横幅だけ時間を要する。サーバ1号72は、サーバ2号73を生成し画像データを部分複製するのに領域82の横幅だけ時間を要する。サーバ2号73は、サーバ3号74を生成し画像データを部分複製するのに領域83の横幅だけ時間を要する。この例は、図17の場合と同じく、サーバ4号が生成できなかった場合である。

【0074】各サーバはサーバの生成及び画像データの部分複製に要した時間だけ、フィルタリングなどの実質的な画像処理を始める時刻が遅れている。つまり、サーバが生成されるごとに各サーバを生成するのにいくらか時間を要し、さらに画像データを部分的に複製するのに時間を要するため、サーバの実質的な画像処理を始める時刻は、サーバIDが大きなサーバほど遅延する。

【0075】上述の処理手続きにより、各サーバのデッドラインよりも、サーバ0号71の処理開始時刻 t_s から実質的な画像処理を始める時刻までの時間が小さいときには、各サーバはハッチで表現された(領域71a, 72a, 73a, 74aに相当する)処理すべき画像データを処理する。よって、サーバ0号71からサーバ3号74まですべてのサーバは、それぞれ各サーバの処理すべき画像データの実質的な処理を行い、処理の結果がサーバ0号に戻される。サーバ1号72が処理した画像データをサーバ0号に戻すのには、領域84の横幅だけ時間を要する。サーバ2号73が処理した画像データをサーバ0号に戻すのには、領域85の横幅だけ時間を要する。サーバ3号74が処理した画像データをサーバ0号にもどすのには、領域86の横幅だけ時間を要する。そして、サーバ0号に集められた処理結果が利用者に戻される。

【0076】次に第5の実施の形態を説明する。以上は、サーバn号は他のサーバの生成について一度しか試みないものであったが、一定量の処理が終了する度に処理すべきデータの未処理部分のデータの処理のために再度サーバの生成を試みる方法も考えられる。例えば、第1の実施の形態の手順の一部を変更して以下のような手順で画像処理を行うことにする。なお、サーバIDの付け方はサーバIDの付け方の規則1によるものとする。

【0077】図19は第5の実施の形態における各サーバの生成及び画像データの部分複製の様子の示す図である。図19においては、サーバ01号91及びサーバ001号92が第1回目のサーバ生成に失敗し、第2回目のサーバ生成には成功した場合の様子を示す。図中のハッチされた部分はサーバ01号91及びサーバ001号92が処理した画像データであることを示している。サーバ0号の処理手続きについては、第1の実施の形態の手続き(図4に示す)とほぼ同じであるため、相違点のみを説明する。

【0078】第5の実施の形態では第1の実施の形態におけるステップS12の処理の内容が異なる。第5の実

施の形態ではこのステップにおいて、時間管理部が、処理開始時刻 t_s 及び最大ライフ・タイム t_l を設定する。ここで、最大ライフ・タイム t_l とは、利用者が画像処理のために待つことのできる時間である。処理開始時刻 t_s と最大ライフ・タイム t_l が設定された時間管理部は、処理開始時刻を t_s とし、処理終了時刻、すなわちサーバ0号が利用者に画像を処理した結果を戻す時刻を t_e とする。また、以後処理開始時刻 t_s から処理終了時刻 t_e までの時間を t_{es} と表記し($t_{es} = t_e - t_s$)、処理開始時刻 t_s から現在の時刻 t_c までの経過時間を t_{cs} と表記する($t_{cs} = t_c - t_s$)。

【0079】サーバ0号の実行時のライフ・タイム(実際の結果的なライフ・タイム)は t_{es} であり、 t_{es} は t_l 以下($t_l \geq t_{es} > 0$)でなければならない。なお、 t_l は利用者がサーバ0号に与えてもよいし、サーバ0号自身が規定値を設定してもよい。また、時間管理部は、 t_l より小さな値をサーバ再生成時間 t_r として設定する。つまり、 $t_l > t_r > 0$ とする。

【0080】図20は第5の実施の形態におけるサーバn号の処理手順を示すフローチャートである。

〔S71〕サーバn号内の時間管理部は、現在の時刻 t_c をサーバn号での処理開始時刻 $t_s(n)$ として設定する。つまり、 $t_s(n) = t_c$ とする。

〔S72〕サーバ生成部は、直ちに自分と同じ処理をするサーバを他の二つの異なる計算機にそれぞれ生成することを試みる。ただし、そのサーバ生成において、処理すべき画像データの一部しか複製しない。ここで、生成した偶数IDのサーバに対して処理すべきデータの前半部分を複製し、生成した奇数IDのサーバに対して処理すべきデータの後半部分を複製する。なお、サーバ生成に失敗した場合、次の手続きに進む。

〔S73〕画像処理部は、処理すべき画像データに対する画像処理を、画像データの先頭から処理すべきデータを最小の実行単位分だけ実行する。

〔S74〕時間管理部は、現在の時刻 t_c を取得する。

〔S75〕時間管理部は、経過時間 $t_{cs} (= t_c - t_s)$ と最大ライフ・タイム t_l とを比較する。経過時間 t_{cs} が最大ライフ・タイム t_l に達していなければステップS76に進み、達していれば自己のプロセスを消滅させる。

〔S76〕生存時間あり、すなわち $t_l \geq t_{cs}$ の場合、時間管理部は、現在の時刻 t_c が $t_s(n) + t_r$ の値より小さいか否かを判断する。現在の時刻 t_c が $t_s(n) + t_r$ の値より小さければステップS72に戻り、そうでなければステップS77に進む。

〔S77〕生存時間あり、すなわち $t_l \geq t_{cs}$ の場合、画像処理結果送出部は、直ちに自己の持つ画像処理結果データをサーバ0号に戻し、自己のプロセスを消滅させる。

【0081】このようにして、サーバ n 号が他のサーバを繰り返し生成すると自己の計算機内にトラブルが発生しても、生成した他のサーバが自己の処理すべき画像データと同じデータを処理してくれる。そのため、サーバ0号は確実に処理済の画像データを受け取ることができる。

【0082】次に第6の実施の形態について説明する。この実施の形態では、あるサーバが他の計算機にサーバを生成した場合に、子供を生むことに例えて1世代増えたと表現することにする。そうすると、前記の実施の形態1又は実施の形態2のサーバIDの付け方の規則においてサーバIDの桁は世代を表現していることになる。つまり、サーバIDが0号であるサーバは第1世代、サーバIDが00号、01号であるサーバはともに第2世代、サーバIDが000号、001号、010号、011号であるサーバはすべて第3世代ということになる。また、第3の実施の形態のサーバIDの付け方の規則においては、サーバIDに1を加えた数が世代数に相当する。

【0083】そして、前述の実施の形態においては世代数に特に制限を設けてはいなかったが、第6の実施の形態では、世代数に制限を設けるようにする。例えば、サーバ自身が自分の世代数をサーバIDから判断し、ある世代数の最大値と同じとなった場合には、そのサーバはさらにサーバを生成することはせず、処理すべき画像データをすべて自ら処理するようにする。

【0084】ところで、以上の実施の形態においては、その処理すべき画像データが任意の画像データである場合を示した。そこで、画像データの内容がその並び方によって特徴がある場合の例を以下に示す。

【0085】例えば、spectral selection方式又はsuccessive approximation方式でプログレッシブ符号化されたJPEG形式の画像データの場合、DCT(Discrete Cosine Transform)係数は周波数方向又はビット方向に段階的符号化されている。そこで、以下の実施の形態によれば、解像度又は階調性の低いおおまかな復号画像から画像処理を行うことを期待できる。したがって、処理した結果を順次利用者に提示すれば、利用者は短時間の内におおまかな画像を見ることができる。

【0086】第7の実施の形態として、DCT係数がspectral selection方式で符号化されているときについて説明する。第7の実施の形態では、第1スキャンから第 p スキャンまでを画像データの前半部分に配置し、第 $p+1$ スキャンから第 n スキャンまでを画像データの後半部分に配置する。つまり、全ブロックについてDCT係数次元の63から32までを画像データの前半部分に配置し、全ブロックについてDCT係数次元の31から0までを画像データの後半部分に配置

する。このように画像データを構成して、第4の実施の形態の方法によって画像処理を行い副画像処理装置から画像処理結果データを受けとるときに、主画像処理装置がそのデータを利用者に戻せば、利用者は、解像度の低い復号画像から画像処理結果を得ることができる。

【0087】図21は第7の実施の形態におけるサーバの生成と画像データの複製状況の例を示す図である。この例では、二つのスキャンで画像データを構成する場合を考える。ここで、画像データ410の前半部分は第1スキャン(第1バンド)411はDCT係数次元の63から32までのデータで構成され、画像データの後半部分は第2スキャン(第2バンド)412はDCT係数次元の31から0までのデータで構成される。第4の実施の形態の場合と同じようにサーバが四つ生成された場合、サーバ0号413はDCT係数次元の63から32までのデータを処理し、サーバ1号414はDCT係数次元の31から16までのデータを処理し、サーバ2号415はDCT係数次元の15から8までのデータを処理し、サーバ3号416はDCT係数次元の7から0までのデータを処理する。よって、サーバIDが大きいサーバほど周波数の低いデータを処理することになる。

【0088】また、図18の実質的な画像処理をする部分を意味するハッチ部分を各サーバで比較すると分かるように、サーバIDが大きいサーバほどその処理結果データをサーバに戻す時点は早い。したがって、主画像処理装置が副画像処理装置から画像処理結果データを受け取るときに、それを利用者に戻せば、利用者は解像度の低い画像から見ることができる。

【0089】次に第8の実施の形態として、DCT係数がsuccessive approximation方式で符号化されているときの処理を説明する。第8の実施の形態では、第1スキャンから第 p スキャンまでを画像データの前半部分に配置し、第 $p+1$ スキャンから第 n スキャンまでを画像データの後半部分に配置する。つまり、全ブロックについてLSB側から、すなわち第0ビットから第3までを画像データの前半部分に配置し、全ブロックについて第4ビットから第7ビットまでを画像データの後半部分に配置する。このように画像データを構成して、第4の実施の形態の方法によって画像処理をすれば、階調のあらい復号画像から画像処理を行うことができる。

【0090】図22は第8の実施の形態におけるサーバの生成と画像データの複製状況の例を示す図である。この例では、二つのスキャンで画像データを構成する場合を考える。ここで、画像データ420の前半部分は第1スキャン(第1バンド)421である第0ビットから第3ビットまで、すなわちLSB側の下位4ビットのデータで構成され、画像データの後半部分は第2スキャン(第2バンド)422で第4ビットから第7ビットまで、すなわちMSB側の上位4ビットのデータで構成さ

れる。第4の実施の形態の場合と同じようにサーバが四つ生成された場合、サーバ0号423は第0ビットから第3ビットまでのデータを処理し、サーバ1号424は第4ビット及び第5ビットのデータを処理し、サーバ2号425は第6ビットのデータを処理し、サーバ3号426は第7ビットのデータを処理する。よって、サーバIDが大きいサーバほど階調のあらいデータを処理することになる。また、図18の実質的な画像処理をする部分を意味するハッチ部分を各サーバで比較すると分かるように、サーバIDが大きいサーバほどその画像処理結果をサーバ0号423に戻す時点は早い。したがって、サーバIDが大きいサーバほど早く階調のあらいデータの処理、例えば逆DCTすなわち復号化などの処理を行った結果を利用者に戻すことができることになる。

【0091】以上は、画像データの分割するときとその画像データを二つに分割する場合について説明したが、さらに分割数を増やして、三分割あるいはさらに分割したn分割等とすることもできる。例えば、第5の実施の形態の場合においてスキャンラインを3以上とし、各スキャンラインごとに画像データを分割することが考えられる。つまり、画像データとしてDCT係数がspectral selection方式で符号化されているときは、全ブロックについてDCT係数次元ごとに分割することもできる。このとき、64個のスキャンラインとなる。一方、画像データとしてDCT係数がsuccessive approximation方式で符号化されているときは、全ブロックについて各ビットごとに分割することもできる。このとき、8個のスキャンラインとなる。

【0092】なお、上記の説明では、1台の計算機で、サーバを一つだけ生成できる場合について説明したが、サーバは単なるプロセスであるため1台の計算機で複数個生成することも可能である。例えば、マルチCPUの計算機では、CPUの数だけサーバを生成するようにしてもよい。

【0093】

【発明の効果】以上説明したように本発明では、処理対象画像データが主画像処理装置に入力されると、主画像処理依頼手段が処理対象画像データの一部の画像処理要求を出力し、その画像処理要求を受け取った副画像処理装置も、受け取った画像データの一部の画像処理要求を出力する。そして、主画像処理実行手段と副画像処理実行手段とがそれぞれ画像処理を実行する。そのため、スケジューリングを管理する機構を設けることなしに分散して画像処理を行うことができる。

【0094】また、副画像処理実行手段は、所定の画像処理を終了するか、あるいは画像処理開始から所定の時間経過した時点で、処理済画像データを主画像処理装置へ転送し、主画像処理実行手段は、処理対象画像データの入力から所定の時間経過した時点で、画像処理結果デ

ータを利用者に戻すようにしたため、利用者が期待する応答時間以内に画像処理の結果を戻すことができる。

【図面の簡単な説明】

【図1】本発明の原理構成図である。

【図2】本発明の第1の実施の形態の概略構成を示す図である。

【図3】計算機の内部構成を示すブロック図である。

【図4】第1の実施の形態のサーバ0号の処理手順を示すフローチャートである。

【図5】第1の実施の形態のサーバn号の処理手順を示すフローチャートである。

【図6】第1の実施の形態におけるサーバIDの付け方の規則を示す図である。

【図7】サーバ001号がサーバの生成に失敗した状況を示す図である。

【図8】サーバ0号の画像処理結果データ受付状況を示す図である。

【図9】画像処理結果データが重複した場合におけるタイミング・チャートである。

【図10】画像処理結果データを受け付けたサーバ0号の処理手順を示す図である。

【図11】第2の実施の形態におけるサーバ0号の処理手順を示すフローチャートである。

【図12】第2の実施の形態におけるサーバIDの付け方を示す図である。

【図13】第2の実施の形態における各サーバ0号の処理すべき画像データ及び処理可能データの両領域の変化を示す図である。

【図14】第3の実施の形態におけるサーバIDの付け方の規則を示す図である。

【図15】第4の実施の形態におけるサーバ0号の処理手順を示すフローチャートである。

【図16】第4の実施の形態におけるサーバn号の処理手順を示すフローチャートである。

【図17】理想的な状況における各サーバが生成されるタイミングと当該サーバで画像データが処理される時間示す図である。

【図18】実際の各サーバが生成されるタイミングと当該サーバで画像データが処理される時間を示す図である。

【図19】第5の実施の形態における各サーバの生成及び画像データの部分複製の様子の示す図である。

【図20】第5の実施の形態におけるサーバn号の処理手順を示すフローチャートである。

【図21】第7の実施の形態におけるサーバの生成と画像データの複製状況の例を示す図である。

【図22】第8の実施の形態におけるサーバの生成と画像データの複製状況の例を示す図である。

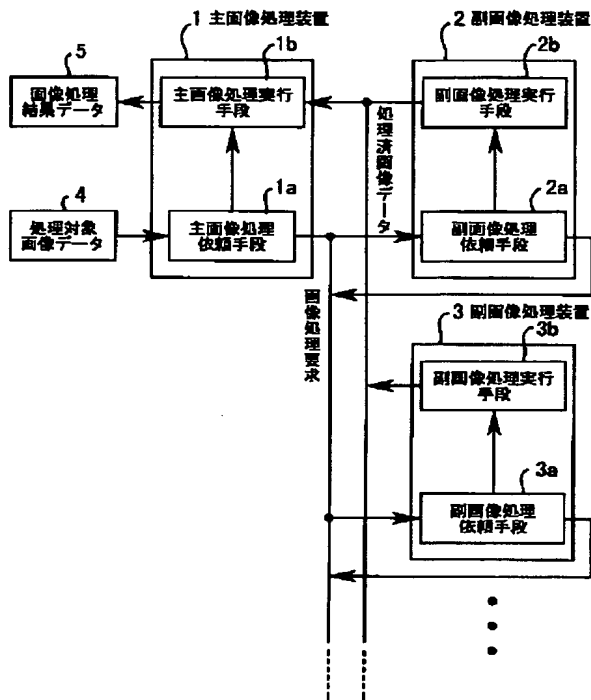
【符号の説明】

1 主画像処理装置

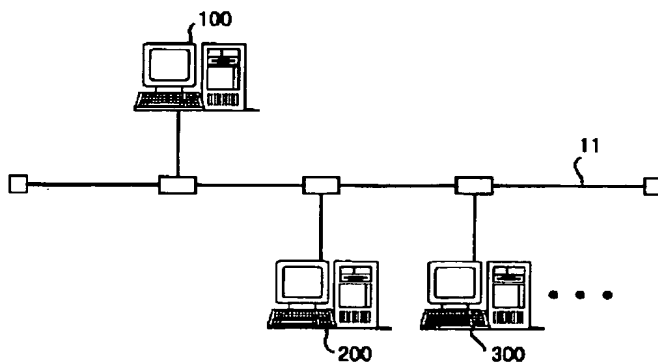
- 1 a 主画像処理依頼手段
1 b 主画像処理実行手段
2, 3 副画像処理装置
2 a, 3 a 副画像処理依頼手段

- 2 b, 3 b 副画像処理実行手段
4 処理対象画像データ
5 画像処理結果データ

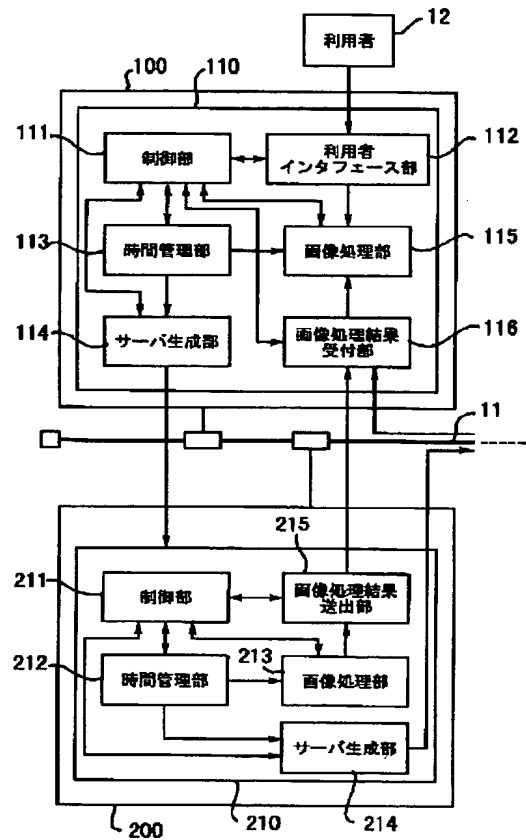
【図1】



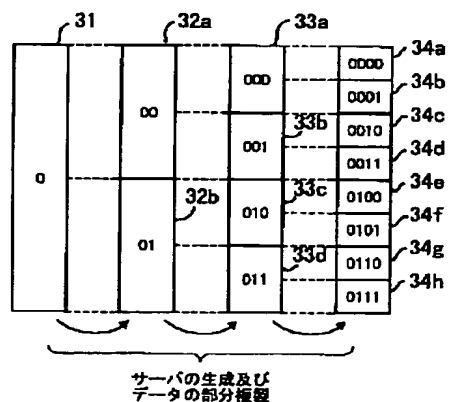
【図2】



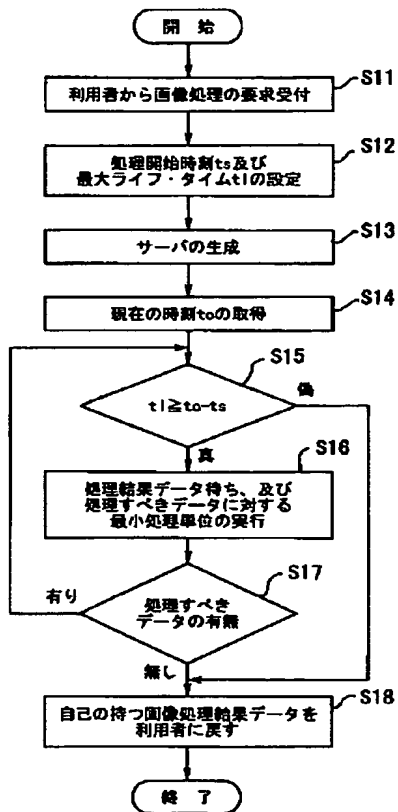
【図3】



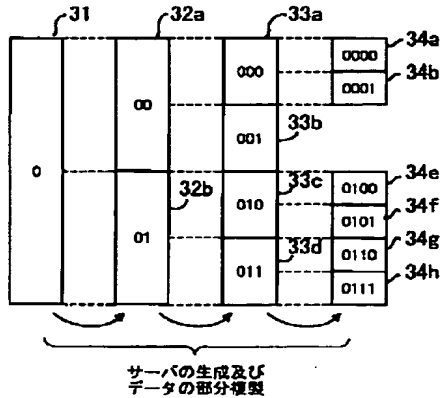
【図6】



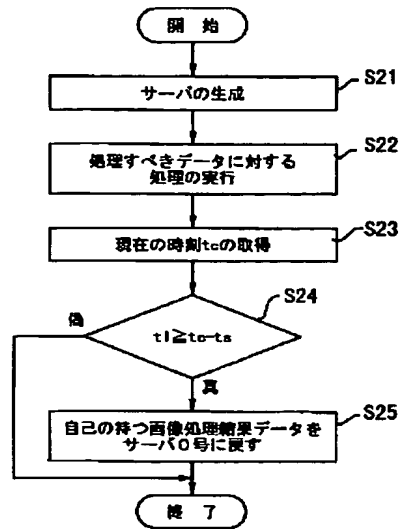
【図4】



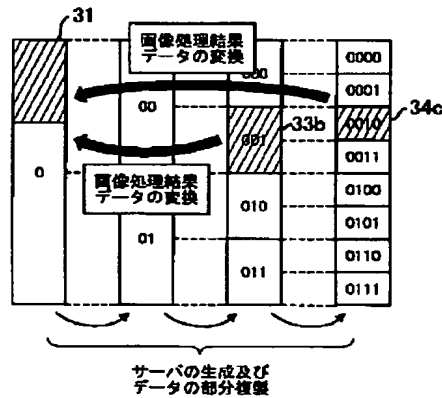
【図7】



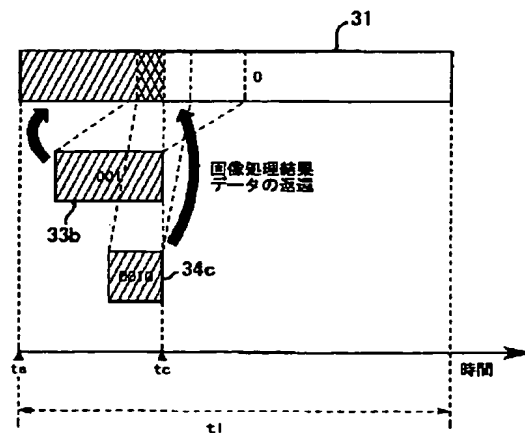
【図5】



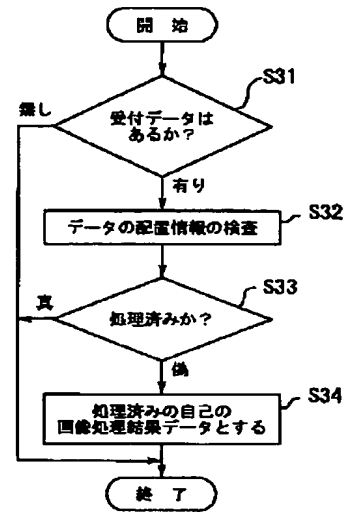
【図8】



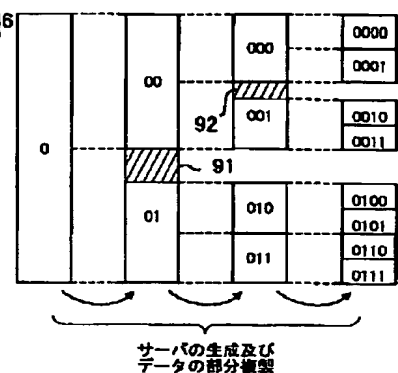
【図9】



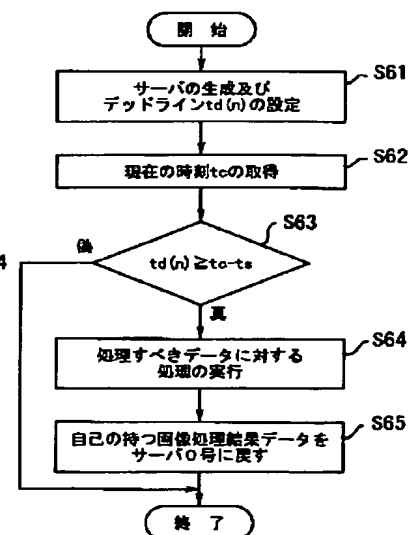
【図10】



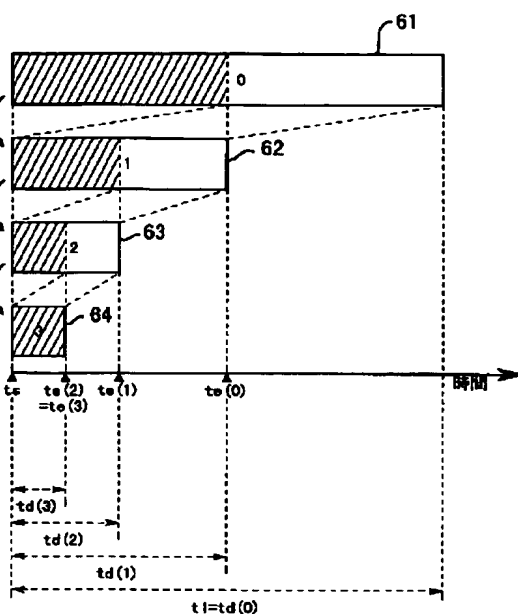
【图 19】



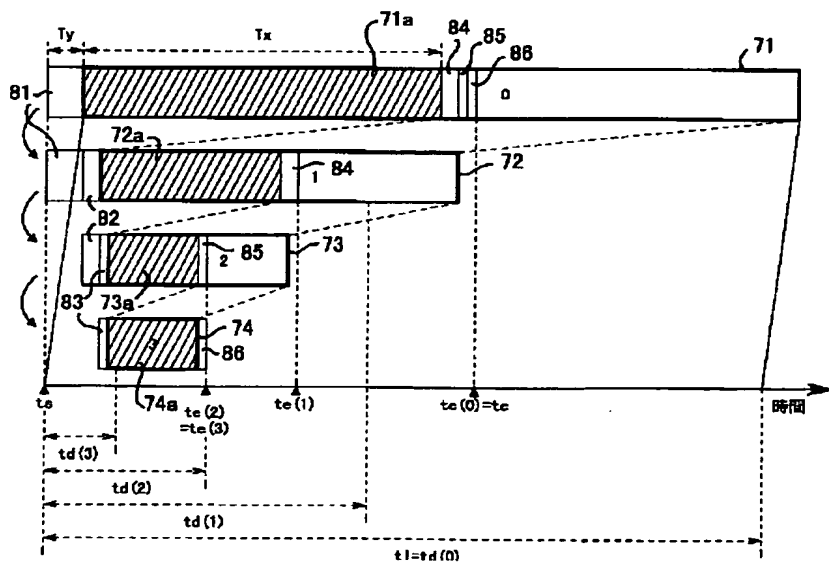
【图16】，



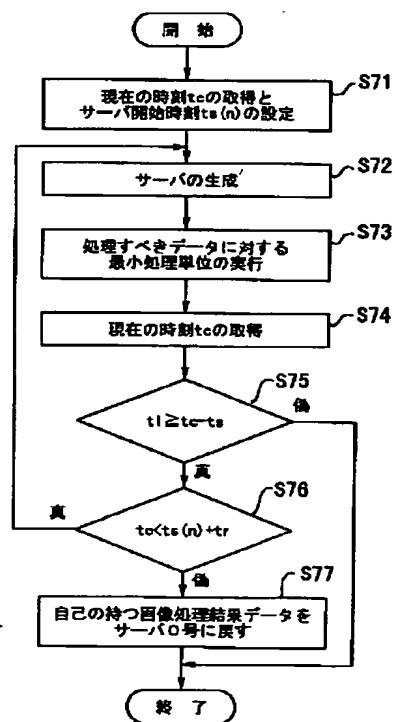
【图 17】



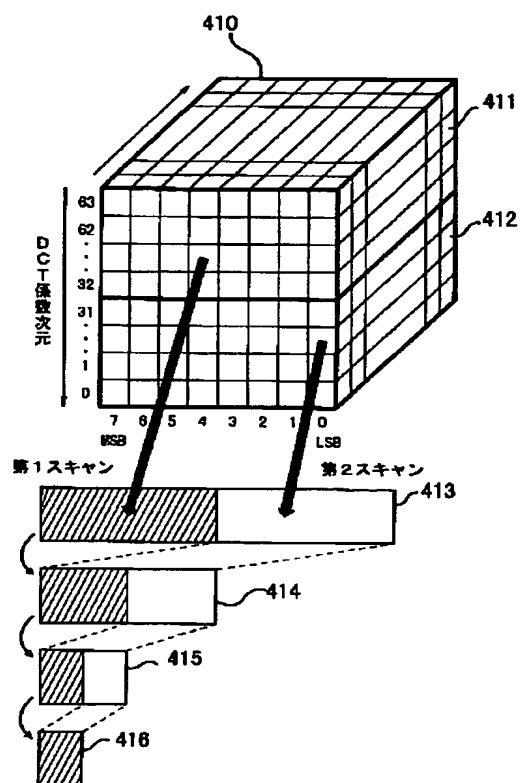
【例18】



【図20】



【図21】



【図22】

